



# Configure Segment Routing for IS-IS Protocol

Integrated Intermediate System-to-Intermediate System (IS-IS), Internet Protocol Version 4 (IPv4), is a standards-based Interior Gateway Protocol (IGP). The Cisco IOS XR software implements the IP routing capabilities described in International Organization for Standardization (ISO)/International Engineering Consortium (IEC) 10589 and RFC 1995, and adds the standard extensions for single topology and mult topology IS-IS for IP Version 6 (IPv6).

This module provides the configuration information used to enable segment routing for IS-IS.

- [Enabling Segment Routing for IS-IS Protocol, on page 1](#)
- [Configuring a Prefix-SID on the IS-IS Enabled Loopback Interface, on page 3](#)
- [Configuring an Adjacency SID, on page 6](#)
- [Configuring Bandwidth-Based Local UCMP, on page 11](#)
- [IS-IS Multi-Domain Prefix SID and Domain Stitching: Example, on page 13](#)

## Enabling Segment Routing for IS-IS Protocol

Segment routing on the IS-IS control plane supports the following:

- IPv4 and IPv6 control plane
- Level 1, level 2, and multi-level routing
- Prefix SIDs for host prefixes on loopback interfaces
- Adjacency SIDs for adjacencies
- MPLS penultimate hop popping (PHP) and explicit-null signaling

This task explains how to enable segment routing for IS-IS.

### Before you begin

Your network must support the MPLS Cisco IOS XR software feature before you enable segment routing for IS-IS on your router.



**Note** You must enter the commands in the following task list on every IS-IS router in the traffic-engineered portion of your network.

## Procedure

	Command or Action	Purpose
<b>Step 1</b>	<b>configure</b> <b>Example:</b>  RP/0/RP0/CPU0:router# configure	Enters mode.
<b>Step 2</b>	<b>router isis <i>instance-id</i></b> <b>Example:</b>  RP/0/RP0/CPU0:router(config)# <b>router isis</b> <b>isp</b>	Enables IS-IS routing for the specified routing instance, and places the router in router configuration mode.  <b>Note</b> You can change the level of routing to be performed by a particular routing instance by using the <b>is-type</b> router configuration command.
<b>Step 3</b>	<b>address-family { ipv4   ipv6 } [ unicast ]</b> <b>Example:</b>  RP/0/RP0/CPU0:router(config-isis)# <b>address-family ipv4 unicast</b>	Specifies the IPv4 or IPv6 address family, and enters router address family configuration mode.
<b>Step 4</b>	<b>metric-style wide [ level { 1   2 } ]</b> <b>Example:</b>  RP/0/RP0/CPU0:router(config-isis-af)# <b>metric-style wide level 1</b>	Configures a router to generate and accept only wide link metrics in the Level 1 area.
<b>Step 5</b>	<b>router-id loopback <i>loopback interface used for prefix-sid</i></b> <b>Example:</b>  RP/0/RP0/CPU0:router(config-isis-af)#router-id loopback0	Configures router ID for each address-family (IPv4/IPv6).  IS-IS advertises the router ID in TLVs 134 (for IPv4 address family) and 140 (for IPv6 address family). Required when traffic engineering is used.
<b>Step 6</b>	<b>segment-routing mpls</b> <b>Example:</b>  RP/0/RP0/CPU0:router(config-isis-af)# <b>segment-routing mpls</b>	Segment routing is enabled by the following actions: <ul style="list-style-type: none"> <li>• MPLS forwarding is enabled on all interfaces where IS-IS is active.</li> <li>• All known prefix-SIDs in the forwarding plain are programmed, with the prefix-SIDs advertised by remote routers or learned through local or remote mapping server.</li> <li>• The prefix-SIDs locally configured are advertised.</li> </ul>
<b>Step 7</b>	<b>exit</b>	

	Command or Action	Purpose
	<b>Example:</b>  RP/0/RP0/CPU0:router(config-isis-af) # <b>exit</b> RP/0/RP0/CPU0:router(config-isis) # <b>exit</b>	
<b>Step 8</b>	Use the <b>commit</b> or <b>end</b> command.	<b>commit</b> —Saves the configuration changes and remains within the configuration session.  <b>end</b> —Prompts user to take one of these actions: <ul style="list-style-type: none"> <li>• <b>Yes</b> — Saves configuration changes and exits the configuration session.</li> <li>• <b>No</b> —Exits the configuration session without committing the configuration changes.</li> <li>• <b>Cancel</b> —Remains in the configuration session, without committing the configuration changes.</li> </ul>

**What to do next**

Configure the prefix SID.

## Configuring a Prefix-SID on the IS-IS Enabled Loopback Interface

A prefix segment identifier (SID) is associated with an IP prefix. The prefix SID is manually configured from the segment routing global block (SRGB) range of labels. A prefix SID is configured under the loopback interface with the loopback address of the node as the prefix. The prefix segment steers the traffic along the shortest path to its destination.

A prefix SID can be a node SID or an Anycast SID. A node SID is a type of prefix SID that identifies a specific node. An Anycast SID is a type of prefix SID that identifies a set of nodes, and is configured with n-flag clear. The set of nodes (Anycast group) is configured to advertise a shared prefix address and prefix SID. Anycast routing enables the steering of traffic toward multiple advertising nodes. Packets addressed to an Anycast address are forwarded to the topologically nearest nodes.

The prefix SID is globally unique within the segment routing domain.

This task explains how to configure prefix segment identifier (SID) index or absolute value on the IS-IS enabled Loopback interface.

**Before you begin**

Ensure that segment routing is enabled on the corresponding address family.

## Procedure

	Command or Action	Purpose
<b>Step 1</b>	<b>configure</b> <b>Example:</b> RP/0/RP0/CPU0:router# configure	Enters mode.
<b>Step 2</b>	<b>router isis <i>instance-id</i></b> <b>Example:</b> RP/0/RP0/CPU0:router (config)# <b>router isis</b> <b>1</b>	Enables IS-IS routing for the specified routing instance, and places the router in router configuration mode. <ul style="list-style-type: none"> <li>You can change the level of routing to be performed by a particular routing instance by using the <b>is-type</b> router configuration command.</li> </ul>
<b>Step 3</b>	<b>interface Loopback <i>instance</i></b> <b>Example:</b> RP/0/RP0/CPU0:router (config-isis)# <b>interface Loopback0</b>	Specifies the loopback interface and instance.
<b>Step 4</b>	<b>address-family { ipv4   ipv6 } [ unicast ]</b> <b>Example:</b> The following is an example for ipv4 address family: RP/0/RP0/CPU0:router (config-isis-if)# <b>address-family ipv4 unicast</b>	Specifies the IPv4 or IPv6 address family, and enters router address family configuration mode.
<b>Step 5</b>	<b>prefix-sid [algorithm <i>algorithm-number</i>]</b> <b>{index <i>SID-index</i>   absolute <i>SID-value</i>}</b> <b>[n-flag-clear] [explicit-null ]</b> <b>Example:</b> RP/0/RP0/CPU0:router (config-isis-if-af)# <b>prefix-sid index 1001</b>  RP/0/RP0/CPU0:router (config-isis-if-af)# <b>prefix-sid absolute 17001</b>	Configures the prefix-SID index or absolute value for the interface. <p>Specify <b>algorithm <i>algorithm-number</i></b> to configure SR Flexible Algorithm. See <a href="#">Enabling Segment Routing Flexible Algorithm</a>.</p> <p>Specify <b>index <i>SID-index</i></b> for each node to create a prefix SID based on the lower boundary of the SRGB + the index.</p> <p>Specify <b>absolute <i>SID-value</i></b> for each node to create a specific prefix SID within the SRGB.</p> <p>By default, the n-flag is set on the prefix-SID, indicating that it is a node SID. For specific prefix-SID (for example, Anycast prefix-SID), enter the <b>n-flag-clear</b> keyword. IS-IS does not set the N flag in the prefix-SID sub Type Length Value (TLV).</p>

	Command or Action	Purpose
		To disable penultimate-hop-popping (PHP) and add explicit-Null label, enter <code>explicit-null</code> keyword. IS-IS sets the <code>E</code> flag in the prefix-SID sub TLV.
<b>Step 6</b>	Use the <code>commit</code> or <code>end</code> command.	<p><b>commit</b>—Saves the configuration changes and remains within the configuration session.</p> <p><b>end</b>—Prompts user to take one of these actions:</p> <ul style="list-style-type: none"> <li>• <b>Yes</b> — Saves configuration changes and exits the configuration session.</li> <li>• <b>No</b> —Exits the configuration session without committing the configuration changes.</li> <li>• <b>Cancel</b> —Remains in the configuration session, without committing the configuration changes.</li> </ul>

Verify the prefix-SID configuration:

```
RP/0/RP0/CPU0:router# show isis database verbose

IS-IS 1 (Level-2) Link State Database
LSPID                LSP Seq Num  LSP Checksum  LSP Holdtime  ATT/P/OL
router.00-00        * 0x0000039b  0xfc27        1079           0/0/0
  Area Address: 49.0001
  NLPID:            0xcc
  NLPID:            0x8e
  MT:               Standard (IPv4 Unicast)
  MT:               IPv6 Unicast                               0/0/0
  Hostname:         router
  IP Address:       10.0.0.1
  IPv6 Address:     2001:0db8:1234::0a00:0001
  Router Cap:       10.0.0.1, D:0, S:0
    Segment Routing: I:1 V:1, SRGB Base: 16000 Range: 8000
    SR Algorithm:
      Algorithm: 0
<...>
  Metric: 0          IP-Extended 10.0.0.1/32
    Prefix-SID Index: 1001, Algorithm:0, R:0 N:1 P:0 E:0 V:0 L:0
<...>
```

### What to do next

Configure the SR-TE policy.

## Configuring an Adjacency SID

An adjacency SID (Adj-SID) is associated with an adjacency to a neighboring node. The adjacency SID steers the traffic to a specific adjacency. Adjacency SIDs have local significance and are only valid on the node that allocates them.

An adjacency SID can be allocated dynamically from the dynamic label range or configured manually from the segment routing local block (SRLB) range of labels.

Adjacency SIDs that are dynamically allocated do not require any special configuration, however there are some limitations:

- A dynamically allocated Adj-SID value is not known until it has been allocated, and a controller will not know the Adj-SID value until the information is flooded by the IGP.
- Dynamically allocated Adj-SIDs are not persistent and can be reallocated after a reload or a process restart.
- Each link is allocated a unique Adj-SID, so the same Adj-SID cannot be shared by multiple links.

Manually allocated Adj-SIDs are persistent over reloads and restarts. They can be provisioned for multiple adjacencies to the same neighbor or to different neighbors. You can specify that the Adj-SID is protected. If the Adj-SID is protected on the primary interface and a backup path is available, a backup path is installed. By default, manual Adj-SIDs are not protected.

Adjacency SIDs are advertised using the existing IS-IS Adj-SID sub-TLV. The S and P flags are defined for manually allocated Adj-SIDs.

```

0 1 2 3 4 5 6 7
+-----+-----+
|F|B|V|L|S|P|  |
+-----+-----+

```

**Table 1: Adjacency Segment Identifier (Adj-SID) Flags Sub-TLV Fields**

Field	Description
S (Set)	This flag is set if the same Adj-SID value has been provisioned on multiple interfaces.
P (Persistent)	This flag is set if the Adj-SID is persistent (manually allocated).

Manually allocated Adj-SIDs are supported on point-to-point (P2P) interfaces.

This task explains how to configure an Adj-SID on an interface.

### Before you begin

Ensure that segment routing is enabled on the corresponding address family.

Use the **show mpls label table detail** command to verify the SRLB range.

## Procedure

	Command or Action	Purpose
<b>Step 1</b>	<b>configure</b> <b>Example:</b> RP/0/RP0/CPU0:router# <b>configure</b>	Enters mode.
<b>Step 2</b>	<b>router isis <i>instance-id</i></b> <b>Example:</b> RP/0/RP0/CPU0:router(config)# <b>router isis</b> <b>1</b>	Enables IS-IS routing for the specified routing instance, and places the router in router configuration mode. <ul style="list-style-type: none"> <li>You can change the level of routing to be performed by a particular routing instance by using the <b>is-type</b> router configuration command.</li> </ul>
<b>Step 3</b>	<b>interface <i>type interface-path-id</i></b> <b>Example:</b> RP/0/RP0/CPU0:router(config-isis)# <b>interface GigabitEthernet0/0/0/7</b>	Specifies the interface and enters interface configuration mode.
<b>Step 4</b>	<b>point-to-point</b> <b>Example:</b> RP/0/RP0/CPU0:router(config-isis-if)# <b>point-to-point</b>	Specifies the interface is a point-to-point interface.
<b>Step 5</b>	<b>address-family { ipv4   ipv6 } [ unicast ]</b> <b>Example:</b> The following is an example for ipv4 address family: RP/0/RP0/CPU0:router(config-isis-if)# <b>address-family ipv4 unicast</b>	Specifies the IPv4 or IPv6 address family, and enters router address family configuration mode.
<b>Step 6</b>	<b>adjacency-sid {index <i>adj-SID-index</i>   absolute <i>adj-SID-value</i> } [protected ]</b> <b>Example:</b> RP/0/RP0/CPU0:router(config-isis-if-af)# <b>adjacency-sid index 10</b> RP/0/RP0/CPU0:router(config-isis-if-af)# <b>adjacency-sid absolute 15010</b>	Configures the Adj-SID index or absolute value for the interface. <p>Specify <b>index <i>adj-SID-index</i></b> for each link to create an Adj-SID based on the lower boundary of the SRLB + the index.</p> <p>Specify <b>absolute <i>adj-SID-value</i></b> for each link to create a specific Adj-SID within the SRLB.</p> <p>Specify if the Adj-SID is <b>protected</b>. For each primary path, if the Adj-SID is protected on the primary interface and a backup path is available, a backup path is installed. By default, manual Adj-SIDs are not protected.</p>

	Command or Action	Purpose
<b>Step 7</b>	Use the <b>commit</b> or <b>end</b> command.	<p><b>commit</b> —Saves the configuration changes and remains within the configuration session.</p> <p><b>end</b> —Prompts user to take one of these actions:</p> <ul style="list-style-type: none"> <li>• <b>Yes</b> — Saves configuration changes and exits the configuration session.</li> <li>• <b>No</b> —Exits the configuration session without committing the configuration changes.</li> <li>• <b>Cancel</b> —Remains in the configuration session, without committing the configuration changes.</li> </ul>

Verify the Adj-SID configuration:

```
RP/0/RP0/CPU0:router# show isis segment-routing label adjacency persistent
Mon Jun 12 02:44:07.085 PDT

IS-IS 1 Manual Adjacency SID Table

15010 AF IPv4
    GigabitEthernet0/0/0/3: IPv4, Protected 1/65/N, Active
    GigabitEthernet0/0/0/7: IPv4, Protected 2/66/N, Active

15100 AF IPv6
    GigabitEthernet0/0/0/3: IPv6, Not protected 255/255/N, Active
```

Verify the labels are added to the MPLS Forwarding Information Base (LFIB):

```
RP/0/RP0/CPU0:router# show mpls forwarding labels 15010
Mon Jun 12 02:50:12.172 PDT
Local  Outgoing  Prefix          Outgoing      Next Hop      Bytes
Label  Label       or ID          Interface     Interface     Switched
-----
15010  Pop          SRLB (idx 10)  Gi0/0/0/3    10.0.3.3     0
      Pop          SRLB (idx 10)  Gi0/0/0/7    10.1.0.5     0
      16004        SRLB (idx 10)  Gi0/0/0/7    10.1.0.5     0                (!)
      16004        SRLB (idx 10)  Gi0/0/0/3    10.0.3.3     0                (!)
```

### What to do next

Configure the SR-TE policy.

## Manually Configure a Layer 2 Adjacency SID

Typically, an adjacency SID (Adj-SID) is associated with a Layer 3 adjacency to a neighboring node, to steer the traffic to a specific adjacency. If you have Layer 3 bundle interfaces, where multiple physical interfaces



form a bundle interface, the individual Layer 2 bundle members are not visible to IGP; only the bundle interface is visible.

You can configure a Layer 2 Adj-SID for the individual Layer 2 bundle interfaces. This configuration allows you to track the availability of individual bundle member links and to verify the segment routing forwarding over the individual bundle member links, for Operational Administration and Maintenance (OAM) purposes.

A Layer 2 Adj-SID can be allocated dynamically or configured manually.

- IGP dynamically allocates Layer 2 Adj-SIDs from the dynamic label range for each Layer 2 bundle member. A dynamic Layer 2 Adj-SID is not persistent and can be reallocated as the Layer 3 bundle link goes up and down.
- Manually configured Layer 2 Adj-SIDs are persistent if the Layer 3 bundle link goes up and down. Layer 2 Adj-SIDs are allocated from the Segment Routing Local Block (SRLB) range of labels. However, if the configured value of Layer 2 Adj-SID does not fall within the available SRLB, a Layer 2 Adj-SID will not be programmed into forwarding information base (FIB).

### Restrictions

- Adj-SID forwarding requires a next-hop, which can be either an IPv4 address or an IPv6 address, but not both. Therefore, manually configured Layer 2 Adj-SIDs are configured per address-family.
- Manually configured Layer 2 Adj-SID can be associated with only one Layer 2 bundle member link.
- A SID value used for Layer 2 Adj-SID cannot be shared with Layer 3 Adj-SID.
- SR-TE using Layer 2 Adj-SID is not supported.

This task explains how to configure a Layer 2 Adj-SID on an interface.

### Before you begin

Ensure that segment routing is enabled on the corresponding address family.

Use the **show mpls label table detail** command to verify the SRLB range.

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	<b>configure</b> <b>Example:</b> RP/0/RP0/CPU0:router# <code>configure</code>	Enters mode.
<b>Step 2</b>	<b>segment-routing</b> <b>Example:</b> RP/0/RP0/CPU0:Router(config)# <b>segment-routing</b>	Enters segment routing configuration mode.
<b>Step 3</b>	<b>adjacency-sid</b> <b>Example:</b>	Enters adjacency SID configuration mode.

	Command or Action	Purpose
	RP/0/RP0/CPU0:Router(config-sr)# adjacency-sid	
<b>Step 4</b>	<b>interface</b> <i>type interface-path-id</i>  <b>Example:</b> RP/0/RP0/CPU0:Router(config-sr-adj)# interface GigabitEthernet0/0/0/3	Specifies the interface and enters interface configuration mode.
<b>Step 5</b>	<b>address-family</b> { <b>ipv4</b>   <b>ipv6</b> } [ <b>unicast</b> ]  <b>Example:</b> RP/0/RP0/CPU0:Router(config-sr-adj-intf)# address-family ipv4 unicast	Specifies the IPv4 or IPv6 address family, and enters router address family configuration mode.
<b>Step 6</b>	<b>l2-adjacency sid</b> { <b>index</b> <i>adj-SID-index</i>   <b>absolute</b> <i>adj-SID-value</i> } [ <b>next-hop</b> { <i>ipv4_address</i>   <i>ipv6_address</i> } ]  <b>Example:</b> RP/0/RP0/CPU0:Router(config-sr-adj-intf-af)# l2-adjacency sid absolute 15015 next-hop 10.1.1.4	Configures the Adj-SID index or absolute value for the interface.  Specify <b>index</b> <i>adj-SID-index</i> for each link to create an Adj-SID based on the lower boundary of the SRLB + the index.  Specify <b>absolute</b> <i>adj-SID-value</i> for each link to create a specific Adj-SID within the SRLB.  For point-to-point interfaces, you are not required to specify a next-hop. However, if you do specify the next-hop, the Layer 2 Adj-SID will be used only if the specified next-hop matches the neighbor address.  For LAN interfaces, you must configure the next-hop IPv4 or IPv6 address. If you do not configure the next-hop, the Layer 2 Adj-SID will not be used for LAN interface.
<b>Step 7</b>	Use the <b>commit</b> or <b>end</b> command.	<b>commit</b> —Saves the configuration changes and remains within the configuration session.  <b>end</b> —Prompts user to take one of these actions:  • <b>Yes</b> — Saves configuration changes and exits the configuration session.  • <b>No</b> —Exits the configuration session without committing the configuration changes.  • <b>Cancel</b> —Remains in the configuration session, without committing the configuration changes.
<b>Step 8</b>	<b>end</b>	

	Command or Action	Purpose
<b>Step 9</b>	<b>router isis</b> <i>instance-id</i> <b>Example:</b> RP/0/RP0/CPU0:Router(config)# <b>router isis isp</b>	Enables IS-IS routing for the specified routing instance, and places the router in router configuration mode.
<b>Step 10</b>	<b>address-family</b> { <b>ipv4</b>   <b>ipv6</b> } [ <b>unicast</b> ] <b>Example:</b> RP/0/RP0/CPU0:Router(config-isis)# <b>address-family ipv4 unicast</b>	Specifies the IPv4 or IPv6 address family, and enters router address family configuration mode.
<b>Step 11</b>	<b>segment-routing bundle-member-adj-sid</b> <b>Example:</b> RP/0/RP0/CPU0:Router(config-isis-af)# <b>segment-routing bundle-member-adj-sid</b>	Programs the dynamic Layer 2 Adj-SIDs, and advertises both manual and dynamic Layer 2 Adj-SIDs. <b>Note</b> This command is not required to program manual L2 Adj-SID, but is required to program the dynamic Layer 2 Adj-SIDs and to advertise both manual and dynamic Layer 2 Adj-SIDs.

Verify the configuration:

```
Router# show mpls forwarding detail | i "Pop|Outgoing Interface|Physical Interface"
Tue Jun 20 06:53:51.876 PDT
. . . .
15001 Pop          SRLB (idx 1)      BE1          10.1.1.4      0
    Outgoing Interface: Bundle-Ether1 (ifhandle 0x000000b0)
    Physical Interface: GigabitEthernet0/0/0/3 (ifhandle 0x000000b0)
```

```
Router# show running-config segment-routing
Tue Jun 20 07:14:25.815 PDT
segment-routing
 adjacency-sid
  interface GigabitEthernet0/0/0/3
  address-family ipv4 unicast
  12-adjacency-sid absolute 15015
  !
  !
  !
  !
```

## Configuring Bandwidth-Based Local UCMP

Bandwidth-based local Unequal Cost Multipath (UCMP) allows you to enable UCMP functionality locally between Equal Cost Multipath (ECMP) paths based on the bandwidth of the local links.

Bandwidth-based local UCMP is performed for prefixes, segment routing Adjacency SIDs, and Segment Routing label cross-connects installed by IS-IS, and is supported on any physical or virtual interface that has a valid bandwidth.

For example, if the capacity of a bundle interface changes due to the link or line card up/down event, traffic continues to use the affected bundle interface regardless of the available provisioned bundle members. If some bundle members were not available due to the failure, this behavior could cause the traffic to overload the bundle interface. To address the bundle capacity changes, bandwidth-based local UCMP uses the bandwidth of the local links to load balance traffic when bundle capacity changes.

### Before you begin

#### Procedure

	Command or Action	Purpose
<b>Step 1</b>	<b>configure</b> <b>Example:</b> <pre>RP/0/RP0/CPU0:router# configure</pre>	Enters mode.
<b>Step 2</b>	<b>router isis <i>instance-id</i></b> <b>Example:</b> <pre>RP/0/RP0/CPU0:router(config)# router isis 1</pre>	Enables IS-IS routing for the specified routing instance, and places the router in router configuration mode.  You can change the level of routing to be performed by a particular routing instance by using the <b>is-type</b> router configuration command.
<b>Step 3</b>	<b>address-family { ipv4   ipv6 } [ unicast ]</b> <b>Example:</b> The following is an example for ipv4 address family: <pre>RP/0/RP0/CPU0:router(config-isis)# address-family ipv4 unicast</pre>	Specifies the IPv4 or IPv6 address family, and enters IS-IS address family configuration mode.
<b>Step 4</b>	<b>apply-weight ecmp-only bandwidth</b> <b>Example:</b> <pre>RP/0/RP0/CPU0:router(config-isis-af)# apply-weight ecmp-only bandwidth</pre>	Enables UCMP functionality locally between ECMP paths based on the bandwidth of the local links.
<b>Step 5</b>	Use the <b>commit</b> or <b>end</b> command.	<b>commit</b> —Saves the configuration changes and remains within the configuration session. <b>end</b> —Prompts user to take one of these actions: <ul style="list-style-type: none"><li>• <b>Yes</b> — Saves configuration changes and exits the configuration session.</li></ul>

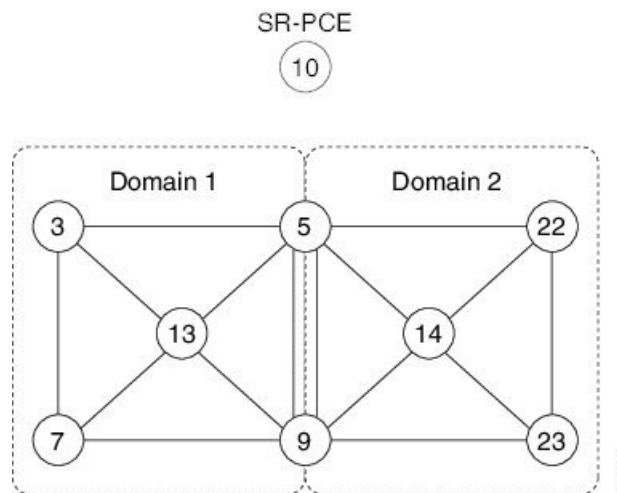
	Command or Action	Purpose
		<ul style="list-style-type: none"> <li>• <b>No</b> —Exits the configuration session without committing the configuration changes.</li> <li>• <b>Cancel</b> —Remains in the configuration session, without committing the configuration changes.</li> </ul>

## IS-IS Multi-Domain Prefix SID and Domain Stitching: Example

IS-IS Multi-Domain Prefix SID and Domain Stitching allows you to configure multiple IS-IS instances on the same loopback interface for domain border nodes. You specify a loopback interface and prefix SID under multiple IS-IS instances to make the prefix and prefix SID reachable in different domains.

This example uses the following topology. Node 5 and 9 are border nodes between two IS-IS domains (Domain1 and Domain2). Node 10 is configured as the Segment Routing Path Computation Element (SR-PCE).

Figure 1: Multi-Domain Topology



### Configure IS-IS Multi-Domain Prefix SID

Specify a loopback interface and prefix SID under multiple IS-IS instances on each border node:

```

Example: Border Node 5
router isis Domain1
 interface Loopback0
  address-family ipv4 unicast
  prefix-sid absolute 16005

router isis Domain2
 interface Loopback0
  address-family ipv4 unicast
  prefix-sid absolute 16005
    
```

**Example: Border Node 9**

```
router isis Domain1
 interface Loopback0
  address-family ipv4 unicast
  prefix-sid absolute 16009
```

```
router isis Domain2
 interface Loopback0
  address-family ipv4 unicast
  prefix-sid absolute 16009
```

Border nodes 5 and 9 each run two IS-IS instances (Domain1 and Domain2) and advertise their Loopback0 prefix and prefix SID in both domains.

Nodes in both domains can reach the border nodes by using the same prefix and prefix SID. For example, Node 3 and Node 22 can reach Node 5 using prefix SID 16005.

## Configure Common Router ID

On each border node, configure a common TE router ID under each IS-IS instance:

**Example: Border Node 5**

```
router isis Domain1
 address-family ipv4 unicast
  router-id loopback0
```

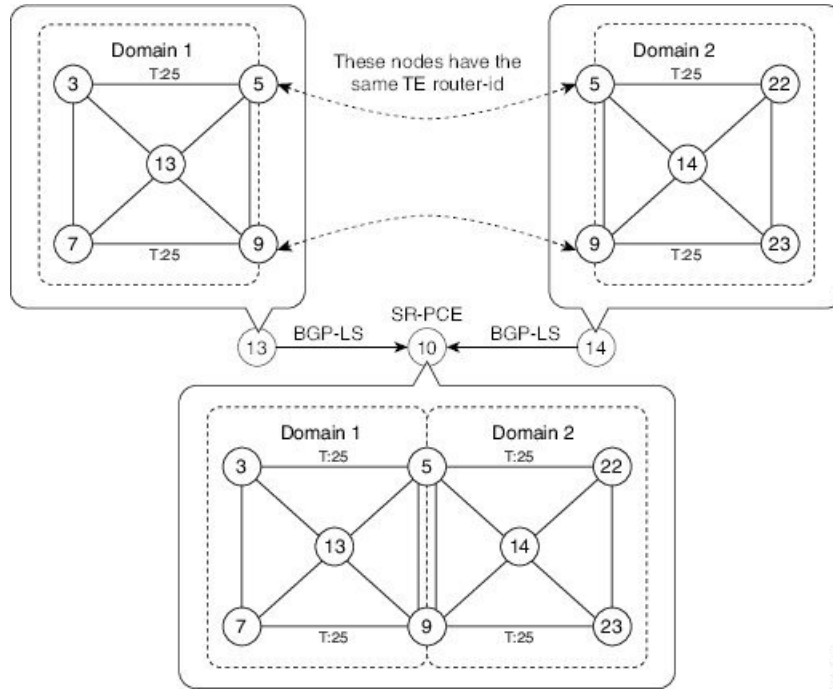
```
router isis Domain2
 address-family ipv4 unicast
  router-id loopback0
```

**Example: Border Node 9**

```
router isis Domain1
 address-family ipv4 unicast
  router-id loopback0
```

```
router isis Domain2
 address-family ipv4 unicast
  router-id loopback0
```

## Distribute IS-IS Link-State Data



Configure BGP Link-state (BGP-LS) on Node 13 and Node 14 to report their local domain to Node 10:

**Example: Node 13**

```
router isis Domain1
  distribute link-state instance-id instance-id
```

**Example: Node 14**

```
router isis Domain2
  distribute link-state instance-id instance-id
```

Link-state ID starts from 32. One ID is required per IGP domain. Different domain IDs are essential to identify that the SR-TE TED belongs to a particular IGP domain.

Nodes 13 and 14 each reports its local domain in BGP-LS to Node 10.

Node 10 identifies the border nodes (Nodes 5 and 9) by their common advertised TE router ID, then combines (stitches) the domains on these border nodes for end-to-end path computations.

